Fingerprint Based Filters (XOR)

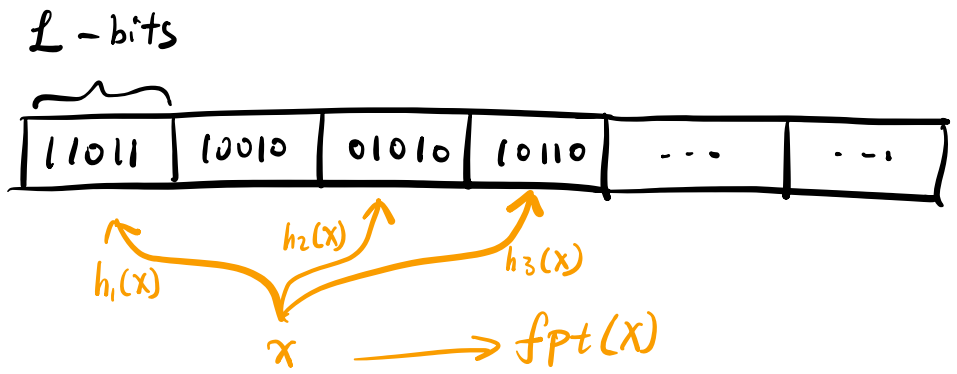


Table code for x: 11011 ⊕ 01010 ⊕ 10110

= 01111

fingerprint function fpt(value) → L bit long code

If fpt(x) = table code for x, i.e. if fpt(x) = 01111,

⇒ probably $x \in$ Table / Set.

Good:

Given a value x, check if $x \in$ Set

Check

PEELING Algo of XOR / BFF is important

Read these ↓ and learn how the set is created.

https://web.stanford.edu/class/archive/cs/cs166/cs166.1216/lectures/13/Slides13.pdf

https://stackoverflow.com/questions/67527507/what-is-an-xor-filter

https://stackoverflow.com/questions/73410580/what-is-a-binary-fuse-filter

Key-Value Filter.

Goal: Works as a Key-Value map but with configurable false positive rate.

F is the "filter" of size N

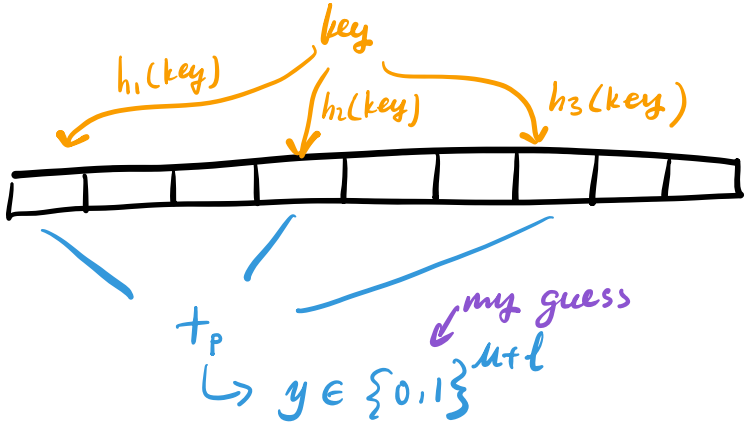
$H = \{h_1, h_2, \ldots, h_k\}$. (example: $k \in \{3,4\}$)

- $F.check(key, H)$ returns:

  $\{F[h_1(key)], \ldots, F[h_k(key)]\}$

- $F.reconstruct(key, H)$:
  - Gets $\{F[h_1(key)], \ldots, F[h_k(key)]\} \leftarrow F.check(key, H)$
  - returns $\bigcirc_{i=1}^{k} F[h_i(key)] \equiv hash(key) \| value$.

If reconstructed result has a correct hash(key), then value is likely correct.



Essence :

KV filter is a fingerprint-based filter (Bloom filter won't work)

such that the value inside the filter is not only used for membership testing. but also for storing information — values.

Instead of asking " is x in this set ? If so, show me the proof ($\bigcirc_{i=1}^{k} F[h_i(x)]$ = $fpt(x)$)

KV-filter asks: " is (key, x) in this set ?

If so, show me the proof ($\bigcirc_{i=1}^{k} F[h_i(key)]$) = $fpt(key, x)$

Since $fpt(key, x)$ is carefully designed, where

$fpt(key, x) = hash(key) \| x$,